State of the OpenDaylight Union

Lisa Caywood, OpenDaylight/Linux Foundation
Colin Dixon, Brocade/OpenDaylight
Outline

- Introduction to OpenDaylight
- Major Use Cases
  - OpenStack integration for cloud
  - CORD and ROBO
- OpenDaylight Architecture
OpenDaylight

Network programmability & automation--
Carriers □ Enterprises □ Universities □
Smart cities and utilities

- Founded 2013 - most mature open networking project
- Most widely deployed OSS NW platform
- >2000 contributors--vendors, users, independents
- 60% of new projects from users
- 5k+ members in global community
Foundation of Open Networking

“OpenDaylight fundamentally changed the Linux Foundation’s world. It’s been wildly successful. It’s the de facto standard open source SDN controller for the industry today.”

- Dave Ward, Cisco CTO

*SDxCentral, 9/7/16
The Open Networking Stack

Automation of Network + Infrastructure + Cloud + Apps + IOT
Evolution of Project Focus

**Common SDN toolchains**
Net Virtualization + SFC:
- OF + OVSDB + OVS/FD.io

Mgmt plane programmability:
- BGP + PCEP + MPLS + NETCONF

**App developer tooling**
- YANG-IDE toolkit
- NetIDE for cross-OSS controller interoperability
- NeXt UI toolkit
- “Singleton app” HA
- Documentation

**Operational tooling**
- Cardinal health monitoring
- Data analytics (TSDR & Centinel)
- OCP (Open radio I/F)
- Documentation

**Integration - industry frameworks**
- OPNFV
- OpenStack enhancement
- CORD/vCO
- ECOMP
- ONF/Atrium
OpenDaylight Carbon Release

- Targeting 5/11/2017

Planned Major Features

- Federated, multi-site OpenStack support
  - Builds on geographic replication and disaster recovery
- Improved robustness and performance of clustering
  - Transaction pipelining
  - Automatic transaction recovery
- Final, RFC version of RESTCONF
- Karaf 4.0
Architecture is Destiny

Modularity == Flexibility
Think Beyond the Controller

Product

Enabling solution component
Graphical User Interface Application and Toolkit (DLUX / NeXT UI)

Independent Network Applications

AAA Authorization Filter

OpenDaylight APIs REST/RESTCONF/NETCONF/AMQP

Control Plane Functions
- AAA
- Host Tracker
- Infrastructure Utilities
- L2 Switch
- LISP Service
- Link Aggregation Control Protocol
- OpenFlow Forwarding Rules Manager
- OpenFlow Stats Manager
- OpenFlow Switch Manager
- Topology Processing

Embedded Controller Applications
- Atrium Router
- Cardinal
- Contention – Streaming Data
- DOCSIS Abstraction
- Eman
- Genius
- NAT Application
- NetIDE
- NetVirt
- Neutron Northbound
- OVSDB Neutron
- SN Integration Aggregator
- Service Function Chaining
- Time Series Data Repository
- Unified Secure Channel Mgr
- User Network Interface Mgr
- Virtual Tenant Network Mgr

Network Abstractions (Policy/Intent)
- ALTO Protocol Manager
- Fabric as a Service
- Group Based Policy Service
- NEMO
- Network Intent Composition
- NetVirt

Data Store (Config & Operational)

Messaging (Notifications / RPCs)

Southbound Interfaces & Protocol Plugins

OpenFlow Enabled Devices

Open vSwitches

Additional Virtual & Physical Devices

Data Plane Elements (Virtual Switches, Physical Device Interfaces)
OpenDaylight is glue for open IT

The glue that holds together L2-7 networking functions

Coordinates physical and virtual resources
  • Network, compute, storage

Network virtualization with service function chaining
  • OpenStack-based network virtualization
  • Virtualized Central Office (vCO; AKA CORD)
OpenDaylight Architecture

What are the major components?
Control Plane Functions
- AAA
- Host Tracker
- Infrastructure Utilities
- L2 Switch
- LISP Service
- Link Aggregation Control Protocol
- OpenFlow Forwarding
- Rules Manager
- OpenFlow Stats Manager
- OpenFlow Switch Manager
- Topology Processing

Embedded Controller Applications
- Atrium Router
- Cardinal
- Contention – Streaming Data
- Hdr
- Controller Shield
- Devs Discovery, ID & Mgmt
- DOC/SIS Abstraction
- Eman
- Genius
- NAT Application
- NetIDE
- NetVirt
- Neutron Northbound
- OVSDB Neutron
- SN Integration Aggregator
- Service Function Chaining
- Time Series Data Repository
- Unified Secure Channel Mgr
- User Network Interface Mgr
- Virtual Tenant Network Mgr

Network Abstractions (Policy/Intent)
- ALTO Protocol Manager
- Fabric as a Service
- Group Based Policy Service
- NEMO
- Network Intent Composition
- NetVirt

Graphical User Interface Application and Toolkit (DLUX / NeXT UI)
- Independent Network Applications
- AAA Authorization Filter

OpenDaylight APIs REST/RESTCONF/NETCONF/AMQP

Data Store (Config & Operational)

Service Abstraction Layer/Core

Messaging (Notifications / RPCs)

Controller Platform Services/Applications

Southbound Interfaces & Protocol Plugins

OpenFlow Enabled Devices

Open vSwitches

Additional Virtual & Physical Devices

Data Plane Elements (Virtual Switches, Physical Device Interfaces)
OpenDaylight with OpenStack
OpenStack/OpenDaylight Integration

- Multiple Neutron implementations
- Target different use cases, southbound drivers
  - FD.io/VPP
  - OVS
  - Open Overlay Router (née LISPmob)
- Provide distributed implementations of scalable network virtualization for OpenStack
OpenStack/OpenDaylight Integration

- L2: ML2 plugin
- L3: ODL L3 plugin
- services
  - FWaaS
  - L2Gateway
  - QoS
  - LBaaS
  - BGPVPN
  - networking-sfc
  - trunk
OpenStack and OpenDaylight Integration

- Management Network
- Data Network
- Public Network

Network/Control Node
- Neutron
  - ML2 Plugin
  - Networking-odl
- ML2 DB

Controller Node
- OpenDaylight*
- Neutron Northbound
- ovsdb/NetVirt
- Yang Model

Compute Node
- VM
- OVS

Networking Node
- DHCP Agent
- OVS

Internet
- Router
OpenDaylight in vCO and ROBO
“We built a CORD system using a set of components...that are open source and generally available — OpenStack, OpenDaylight controllers, Intel software toolkits. Just because you’re doing CORD doesn’t mean you have to do ONOS CORD.”

—Adam Dunstan, CenturyLink VP of SDN and NFV engineering

CenturyLink Surges Forward with Its Own Version of CORD

CenturyLink is on a tight deadline with its commitment to fully virtualize its IP core network by the end of 2019. As part of that, the service provider is emulating some of the open source communities’ work to create a Central Office Re-architected as a Data Center (CORD).

What does a Central Office do?

• **Subscriber management capabilities:** Gateway, authentication and authorization, event and subscriber information logging
• **Optical Line Termination (OLT) for PON/GPON** (Passive Optical Net.)
• **Service functions:** self-service portals, NAT, FW, routing, IP addr mgmt, QoS, quotas, video caching, mail and file stores

A Virtualized Central Office (vCO):
• Uses general-purpose compute, storage and network capabilities to deliver the above services
• Added agility (spin up VMs vs. rack and stack hardware)
• Cost savings (via increased automation and commodity servers)
vCO Data Center Architecture

Physical elements are divided into

– Network: provides fabric/underlay
– Servers: provides computer/storage for VNFs

Fabric/Underlay (Network)

Servers/VNFs (Compute, Storage)
Controllers and orchestrators use overlay networks to form service chains of VNFs
vCO Data Center Software Architecture

**OSS/BSS**

- **Policy** (NIC, NEMO, GBP, Neutron)
- **VNFO** (ECOMP, Open-O, OSM, …)
- **VNF Spec** (TOSCA)
- **VNF Catalog**
- **VNFM** (Tacker, Cloudify, …)
- **VIM** (OpenStack, Kubernetes, …)

**Fabric/Underlay (Network)**

- SDN Controller (OpenDaylight)

**Servers/VNFs (Compute, Storage)**

Service Chaining

Overlay Network

Fabric

Policy (NIC, NEMO, GBP, Neutron)
ROBO: Using vCO Blueprint in Enterprises

- vCO for Enterprises to provide for Remote/Branch offices
  - Maybe offered by ISPs as a service
  - Integrating with public cloud will likely involved some form of vCO (either aaS or Enterprise-deployed)
  - Hybrid Cloud will almost certainly involve vCO
Architectural Drill-Down
Core Architecture

Model-Driven Service Abstraction Layer (MD-SAL)

Notifications
RPCs

Data

Controllers in a Cluster

App/Service
Plugin

YANG Models
ODL is a YANG-based µ-services platform
What is YANG?

- Data modeling language for NETCONF
  - RFC 6020

- Great, what is NETCONF?
  - Think of it as an SNMP replacement with nice features
  - YANG models ~= SNMP MIBs

- OK, fine, but what is YANG?
What is YANG?

- Three core abstractions
  - Data
  - RPCs (just data in and data out)
  - Notifications (just data out)

- So, it’s really all about the data
What does YANG data look like

• container ~= struct
• list ~= map/dictionary
• leaf ~= primitive types
• grouping ~= interface

• Others: typedef, pointers, constraints, etc.
Clustering in OpenDaylight

**Sharded, Replicated Datastore**

- Shard Hash Function: Divide the tree into n shards
- Data tree root
- Shard Hash: X: Service X
  Y: Shard y within Service X
- Shard Layout Algorithm: Place shards on m nodes

**Highly-Available Applications**

- Shards are consistently replicated using RAFT
- Devices/applications can be bound to a single node via Entity Ownership Service
- Automated failover
- Use of “non-voting” cluster members for geographic replication and disaster recovery
Projects & Components in OpenDaylight

Projects

• Categories
  • Kernel
  • Plugins
  • Services
  • Applications
  • Metaproxects
    • https://wiki.opendaylight.org/view/Project_list

Components:

See “A Whirlwind Tour of OpenDaylight”

  o https://youtu.be/ENtQdwsnyjg
  o http://events.linuxfoundation.org/sites/events/files/slides/whirlwind-tour.pdf
Questions?
Thank you